



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/992,823	11/14/2001	Daniel W. Wong	1376-0100520	5879
34456 7590 12/04/2008 LARSON NEWMAN ABEL & POLANSKY, LLP 5914 WEST COURTYARD DRIVE SUITE 200 AUSTIN, TX 78730				
EXAMINER CHAI, LONGBIT				
ART UNIT 2431		PAPER NUMBER		
MAIL DATE 12/04/2008		DELIVERY MODE PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

09/992,823

Applicant(s)

WONG ET AL.

Examiner

LONGBIT CHAI

Art Unit

2431

Period for Reply -- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 20 October 2008.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-47, 49-54, 63 and 64 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 17-30 is/are allowed.
- 6) ☒ Claim(s) 1-16, 31-47, 49-54, 63 and 64 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 14 November 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
- 5) ☐ Notices of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

1. Currently pending claims are 1 – 47, 49 – 54 and 63 – 64.

Response to Arguments

2. Applicant's arguments with respect to the subject matter of the instant claims have been fully considered but are not persuasive.
3. As per claim 1, Applicant asserts (a) the Daellenbach publication is based on a provisional application (hereinafter referred to as the "Daellenbach provisional") filed March 11, 2002, which is subsequent (not prior) to November 14, 2001, the filing date of the present Application. Accordingly, the cited paragraph is not prior art to the present application (Remarks: Page 2 / Last Para) (b) Neither the cited portions, nor any other portion of the Daellenbach provisional, discloses or suggests that the disclose drivers are encrypted in any manner (Remarks: Page 3 / 1st Para) and (c) the Daellenbach provisional fails to disclose "sending a first encrypted routine of a software driver to a peripheral device, wherein the software driver is to interface with the peripheral device" as recited in claim (Remarks: Page 3 / 2nd Para).
Examiner respectfully disagrees with the following rationale.

- (a) "Daellenbach provisional" was filed on 9 March 2001, which is prior to November 14, 2001, the filing date of the instant Application and thereby it is qualified as a prior-art.
- (b) Regarding Applicant's argument "neither the cited portions, nor any other portion of the Daellenbach provisional, discloses or suggests that the disclose drivers are encrypted in any manner" – Examiner respectfully disagrees because Daellenbach Provisional teaches assuming network security requirements are met, a central office can handle firmware

revisions from a single remote location and PC based device drivers that is required by the Administrator can be updated remotely (Provisional SPEC: Page 7, Section 3.3.7: Dynamic Product Updates) and besides, the network security requirements as required by the Administrator also includes encryption techniques that provides the layer structure of secured communications (Provisional SPEC: Page 7, Section 3.4: Security).

- (c) Regarding Applicant's argument "the Daellenbach provisional fails to disclose "sending a first encrypted routine of a software driver to a peripheral device, wherein the software driver is to interface with the peripheral device" – Examiner respectfully disagrees because Daellenbach Provisional teaches "PC based device drivers" (Provisional SPEC: Page 7, Section 3.3.7 / Line 4), wherein Examiner notes (i) it is indeed a software driver that interfaces with a PC peripheral device and (ii) according to Dictionary.com, a peripheral device is merely a computer device that operates separately from the CPU but is connected to it.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1 – 3, 8, 9, 13, 16, 31, 33, 37– 38, 40, 41, 43, 46, 47, 49 – 52, 63 and 64 are rejected under 35 U.S.C. 103(a) as being unpatentable over Daellenbach et al. (U.S. Patent 2003/0168508), in view of Ciacelli (U.S. Patent 6,236,727).

As per claim 1, 31, 40 and 47, Daellenbach teaches a method comprising the steps of:

sending a first encrypted routine of a software driver to a peripheral device

(Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4: (a) the driver software can be download and (b) the encrypted driver software is indeed a driver software (i.e. routine) which is encrypted and can be used at a peripheral device by a software driver after being decrypted. Examiner notes, as per its Provisional application filed on 3/9/2001, Daellenbach Provisional discloses (a) "assuming network security requirements are met, a central office can handle firmware revisions from a single remote location and PC based device drivers required by the Administrator can be updated remotely (Provisional SPEC: Page 7, Section 3.3.7: Dynamic Product Updates) and (b) network security requirements as required by the Administrator includes encryption techniques that provides the layer structure of secured communications (Provisional SPEC: Page 7, Section 3.4: Security). Examiner notes, according to Dictionary.com, a peripheral device is merely a computer device that operates separately from the CPU but is connected to it), **wherein the software driver is to interface with the peripheral device** (Provisional SPEC: Page 7, Section 3.3.7 / Line 4: Daellenbach Provisional teaches "PC based device drivers", wherein Examiner notes (i) it is indeed a software driver that interfaces with a PC peripheral device and (ii) according to Dictionary.com, a peripheral device is merely a computer device that operates separately from the CPU but is connected to it).

However, Daellenbach does not disclose explicitly decrypting, at the peripheral device, the first encrypted routine to generate a plaintext routine.

Ciacelli (combined with Daellenbach) teaches **decrypting, at the peripheral device, the first encrypted routine to generate a plaintext routine** (Ciacelli: Column 5 Line 46 – 48 / Line 43 – 45 / Line 53 – 60: (a) decrypting the downloaded software at the hardware device 27 (b) an encrypted version of actual decryption / encryption software routine (algorithm) matches the first encrypted routine – This is also consistent with dependent claim 2 and 3) &

(Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15: (a) the download encrypted driver software must be decrypted so that it can be used to perform driver function (b) the encrypted driver software after being decrypted is qualified as a plaintext routine).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Ciacelli within the system of Daellenbach because (a) Daellenbach discloses downloading an encrypted device driver remotely to a computer (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4) and (b) Ciacelli teaches a security enhanced mechanism protecting the software / data during the transfer, inside a computer, to various modules / devices within an accessible internal structure by decrypting the encrypted software / data at the device level to avoid exposing the software / data in an unencrypted form (Daellenbach: Column 1 Line 10 – 14 and Column 5 Line 46 – 48).

providing the plaintext routine to the software driver (Daellenbach: see above: a device drive is a software routine used by the software driver at the a peripheral device).

As per claim 49, Daellenbach teaches a method comprising the steps of:

sending a first encrypted data associated with an application to a peripheral device (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4: the encrypted device driver (software routine) and software driver matches the claim language of a first encrypted data and application, respectfully, which are also consistent with claim 50 and 52; besides, Daellenbach teaches (a) the driver software can be download and (b) the encrypted driver software is indeed a driver software (i.e. routine) which is encrypted and can be used at a peripheral device by a software driver after being decrypted. Examiner notes, as per its Provisional application filed on 3/9/2001, Daellenbach Provisional discloses (a)

"assuming network security requirements are met, a central office can handle firmware revisions from a single remote location and PC based device drivers required by the Administrator can be updated remotely (Provisional SPEC: Page 7, Section 3.3.7: Dynamic Product Updates) and (b) network security requirements as required by the Administrator includes encryption techniques that provides the layer structure of secured communications (Provisional SPEC: Page 7, Section 3.4: Security). Examiner notes, according to Dictionary.com, a peripheral device is merely a computer device that operates separately from the CPU but is connected to it), **wherein the application is to interface with the peripheral device** (Provisional SPEC: Page 7, Section 3.3.7 / Line 4: Daellenbach Provisional teaches "PC based device drivers", wherein Examiner notes (i) it is indeed a software driver that interfaces with a PC peripheral device and (ii) according to Dictionary.com, a peripheral device is merely a computer device that operates separately from the CPU but is connected to it).

However, Daellenbach does not disclose explicitly decrypting, at the peripheral device, the first encrypted sensitive to generate a plaintext data.

Ciacelli (combined with Daellenbach) teaches **decrypting, at the peripheral device, the first encrypted sensitive to generate a plaintext data** (Ciacelli: Column 5 Line 46 – 48 / Line 43 – 45 / Line 53 – 60: (a) decrypting the downloaded software at the hardware device 27 (b) an encrypted version of actual decryption / encryption software routine (algorithm) matches the first encrypted routine – This is also consistent with dependent claim 2 and 3) & (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15: (a) the download encrypted driver software must be decrypted so that it can be used to perform driver function (b) the encrypted driver software after being decrypted is qualified as a plaintext routine).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Ciacelli within the system of Daellenbach

because (a) Daellenbach discloses downloading an encrypted device driver remotely to a computer (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4) and (b) Ciacelli teaches a security enhanced mechanism protecting the software / data during the transfer, inside a computer, to various modules / devices within an accessible internal structure by decrypting the encrypted software / data at the device level to avoid exposing the software / data in an unencrypted form (Daellenbach: Column 1 Line 10 – 14 and Column 5 Line 46 – 48).

providing the plaintext data to the application (Daellenbach: see above: (a) a software driver matches the claim language application, which is also consistent with claim 52 (b) a device drive is a software routine used by the software driver at the a peripheral device).

As per claim 2, Daellenbach as modified teaches the first encrypted routine is an encrypted version of an encryption routine (Ciacelli: Column 5 Line 43 – 45 / Line 53 – 60: (a) Daellenbach is relied upon to download an encryption software routine to a device (b) Ciacelli is relied upon teaching the transferred software routine to a peripheral device can be an update of encryption / decryption algorithm routines).

As per claim 3, Daellenbach as modified teaches the first encrypted routine is an encrypted version of a decryption routine (Ciacelli: Column 5 Line 43 – 45 / Line 53 – 60: (a) Daellenbach is relied upon to download an encryption software routine to a device (b) Ciacelli is relied upon teaching the transferred software routine to a peripheral device can be an update of encryption / decryption algorithm routines).

As per claim 8, 33 and 38, Daellenbach as modified teaches sending a decryption code to the peripheral device, where the decryption code is to be used by the peripheral device to decrypt the first encrypted routine (Ciacelli: Column 5 Line 45 – 60).

As per claim 9, Daellenbach as modified teaches removing the plaintext routine (Ciacelli: Column 7 Line 16 – 21).

As per claim 13, Daellenbach as modified teaches selecting the first encrypted routine from a plurality of different encrypted routines, wherein the plurality of different encrypted routines are functionally equivalent (Ciacelli: Column 5 Line 53 – 60: the routines used for the purpose of driver software updates are indeed being selected from a plurality of driver routines with different revisions).

As per claim 16, Daellenbach as modified teaches providing includes storing the plaintext routine in a location in memory accessible by the software driver, and where the location in memory is known to the software driver (Daellenbach: see above: the encrypted routine after being downloaded and decrypted must be, first, stored in the memory somewhere and secondly, must be known to the software device driver so that the driver functions can be performed and executed accordingly by the device driver).

As per claim 41, Daellenbach as modified teaches said first interface and said second interface are implemented using a same interface (Daellenbach : see above: the encrypted

driver software being download can be used by a software driver after being decrypted at the same interface entity).

As per claim 43, Daellenbach as modified teaches the first hardware component and the second component are implemented using a same hardware component (Ciacelli: Column 5 Line 43 – 48: the same hardware component of decryption module to receive and execute the decryption function for encrypted routine).

As per claim 37 and 46, Daellenbach as modified teaches the hardware component is a dedicated hardware component (Ciacelli: Abstract Line 15 – 17, Column 2 Line 55 – 63, Column 5 Line 46 – 48 / Line 43 – 45 / Line 53 – 60).

As per claim 50, Daellenbach as modified teaches the first encrypted data includes an encrypted software routine (Daellenbach : see above).

As per claim 51, Daellenbach as modified teaches the first encrypted data includes an encrypted version of one of: a private encryption key, a private decryption key, a chip ID, and a device ID (Ciacelli: Column 6 Line 42 – 45: the encrypted key and the encrypted data are combined into one data stream and transferred to the peripheral device for the purpose of decrypting usage).

As per claim 52, Daellenbach as modified teaches the application includes a software driver (Daellenbach: see above).

As per claim 63, Daellenbach as modified teaches processing data at the peripheral device using the plaintext routine (Ciacelli: Column 5 Line 46 – 48 / Line 43 – 45 / Line 53 – 60: see above).

As per claim 64, Daellenbach as modified teaches decrypting data at the peripheral device using the plaintext routine (Ciacelli: Column 5 Line 46 – 48 / Line 43 – 45 / Line 53 – 60: see above).

5. Claims 10 – 12, 32, 39, 42 and 54 are rejected under 35 U.S.C. 103(a) as being unpatentable over Daellenbach et al. (U.S. Patent 2003/0168508), in view of Ciacelli (U.S. Patent 6,236,727), in view of Hendricks et al. (U.S. Patent 7,298,851).

As per claim 10, 32, 42 and 54, Daellenbach as modified does not disclose expressly encrypting, at the peripheral device, the plaintext routine to generate a second encrypted routine, where the second encrypted routine is a version of the first encrypted routine.

Hendricks teaches encrypting, at the peripheral device, the plaintext routine to generate a second encrypted routine, where the second encrypted routine is a version of the first encrypted routine (Hendricks: Column 63 Line 22 – 26: secure storage is done on a memory device at the driver-level, where all information (i.e. including the downloaded driver routine disclosed by Carter) stored on the memory storage device is encrypted by a memory device driver prior to being stored on memory storage device);

providing the second encrypted routine to the software driver (see above – merely for storage purpose (i.e. securely store the content) after being downloaded).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Hendricks within the system of Daellenbach as modified because (a) Daellenbach discloses downloading driver routines to a device driver (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4) and (b) Hendricks teaches a secured mechanism to store all information – i.e. including the downloaded content (Hendricks: Column 63 Line 22 – 26).

As per claim 11 and 39, Daellenbach as modified teaches sending a encryption code to the peripheral device (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15), where the encryption code is to be used by the peripheral device to encrypt the plaintext routine (Hendricks: Column 63 Line 22 – 26: encrypting all information prior to storing is indeed including not only data but also routines).

As per claim 12, Daellenbach as modified teaches the second encrypted routine is a modified version of the first encrypted routine (Ciacelli: Column 5 Line 43 – 45 / Line 53 – 60: updated encrypted / decryption algorithm routines) & (Hendricks: Column 63 Line 22 – 26: i.e. storage version).

6. Claims 14 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Daellenbach et al. (U.S. Patent 2003/0168508), in view of Ciacelli (U.S. Patent 6,236,727), in view of Wilson (U.S. Patent 4,520,232).

As per claim 14, Daellenbach as modified does not disclose expressly decrypting includes using a map as a decryption key.

Wilson teaches decrypting includes using a map as a decryption key (Wilson: see for example: Column 2 Line 12 – 24).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Wilson within the system of Daellenbach as modified because (a) Daellenbach discloses downloading an encrypted software routines to a device driver (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4) and (b) Wilson teaches providing an security enhanced encryption mechanism which is not only fast but also inexpensive with increased security strength (Wilson: see for example, Column 1 Line 28 – 34).

As per claim 15, Daellenbach as modified teaches the map includes a texture map (Wilson: see for example, Column 1 Line 28 – 34: the matrix is qualified as a two-dimensional texture map).

7. Claims 4 – 7, 34 – 36, 44 – 45 and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Daellenbach et al. (U.S. Patent 2003/0168508), in view of Ciacelli (U.S. Patent 6,236,727), and in view of Freeman (U.S. Patent 2002/0129374).

As per claim 4, 34 and 53, Daellenbach as modified does not disclose expressly the peripheral device is a graphics chip.

Freeman teaches the hardware device is a graphic chip (Freeman: see for example, Paragraph [0117]).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Freeman within the system of Daellenbach as

modified because (a) Daellenbach discloses downloading a device driver remotely for a peripheral device of a PC-based computer (Daellenbach: Para [0069] Line 10 – 11 / Line 13 – 15 and Provisional Section 3.3.7 and Section 3.4) and (b) Freeman teaches a software driver is needed at a multimedia graphic chip, as one type of peripheral devices, to perform the functions of realizing the MPEG adaptation and processing the video data stream (Freeman: Paragraph [0117] and Figure 7 Element 376 & 388).

As per claim 5 – 6, 35 – 36 and 44 – 45, Daellenbach as modified teaches decrypting is performed by a graphics chip (Ciacelli: see for example: Column 3 Line 25 – 43, Column 5 Line 43 – 60 and Column 2 Line 48 – 50).

Daellenbach as modified does not disclose expressly decrypting is performed by a 3D pipe of the graphics chip.

Daellenbach as modified does not disclose expressly decrypting is performed by a 3D pipe of the graphics chip.

However, Examiner notes Official Notice is taken that a 3D (3-Dimension) engine (or IDCT component) is a well-known series of components inside a video graphic chip.

As per claim 7, Daellenbach as modified teaches decrypting is performed by dedicated encryption hardware of the graphics chip (Ciacelli: see for example: Abstract Line 15 – 17 and Column 2 Line 55 – 63).

Allowable Subject Matter

8. Claims 17 – 30 are allowed.

The following is an examiner's statement of reasons for allowance: The present invention is directed to a method of sending a first encrypted routine of a software driver to a graphics chip, wherein the software driver is to interface with the graphics chip, and where the first encrypted routine is an encrypted version of an encryption routine; decrypting, at the graphics chip, the first encrypted routine to generate a plaintext routine, wherein the plaintext routine is a version of the encryption routine; and storing the plaintext routine in memory in a location known to the software driver.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to LONGBIT CHAI whose telephone number is (571)272-3788. The examiner can normally be reached on Monday-Friday 9:00am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Y. Vu can be reached on 571-272-3859. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Longbit Chai/

Longbit Chai Ph.D.
Patent Examiner
Art Unit 2131
12/01/2008